Integrated Erasure-based Coding for Reliable Multicast Retransmission

Joe Macker
Center for High Assurance Computer Systems,
NRL
macker@itd.nrl.navy.mil

Notes presented to the IRTF RM Meeting
Memphis, TN April 1997

Joe Macker March 1996
**Integrated Erasure-based Coding for Reliable Multicast Retransmission**

Naval Research Laboratory, Center for High Assurance Computer Systems, 4555 Overlook Avenue, SW, Washington, DC, 20375

Approved for public release; distribution unlimited

**Security Classification of:**
- Report: unclassified
- Abstract: unclassified
- This Page: unclassified

**Date Covered:** 00-00-1997 to 00-00-1997

Status: Approved for public release; distribution unlimited

**Abstract:**

No abstract is provided in this report.

**Subject Terms:**
- Integrated Erasure-based Coding
- Reliable Multicast Retransmission
Some Background

K Data Packets

N-K Parity Packets

Lost Packets = Codeword Erasures

J codewords in parallel
e.g., GF(2^8) byte-based symbols
For RS erasure based decoding, any missing
n-k packets in k are decodeable with n-k parity

Erasure-based Correction Methods
1) Send parity with DATA
   Fixed or Variable code rate

2) Request Parity Retransmission
   Send 0 parity on 1st block cycle

3) Hybrid
   Send some parity with data block,
   request more when conditions worsen

Joe Macker March 1996
Integrated Retransmission

- Don’t transmit any parity on 1st cycle
- Modify RM Nack Process
  - report only max lost among receiver group \{block id, max lost\}
  - can still do repair backoff
  - no parsing nacked sequence numbers, block bit maps, etc
- Scaled groups can show large % loss in total
  - single retransmitted parity can repair multiple lost packets
  - significant RM message reduction for uncorrelated loss cases
  - with an integrated retransmission approach you do not have preestimate the amount of parity needed

Joe Macker March 1996
Performance Gain for Uncorrelated Loss

- Single parity packet can repair multiple packets across receiver set
- We can integrate this into Nack processing to improve scaling when uncorrelated loss is anticipated
- Can use this for streaming but there is:
  - delay penalty
  - processing penalty
- Makes a lot of sense for bulk transfer to large groups in one-to-many environments

E.g., Transmit Block (1..20)

\{2,3,8\} \{2,5,7\} \{5,11,15\}

Total Lost Packets = 6
Max Lost Per Receiver = 3
Simple Loss Model Example

1. uncorrelated packet loss
2. Homogeneous loss probability
3. \(N= \) packet block size, \(M= \) receiver group size

Expected Value of 2nd Cycle Repair Retransmissions?

\[\text{NoFEC}(n, m, p) := n \cdot \left[ 1 - (1 - p)^m \right] \]

With parity erasure-based repairing
here’s the pdf of the max among nodes

\[\text{pdfFEC}(n, m, p, k) := \sum_{j=1}^{m} \frac{m!}{(j!)(m-j)!} \cdot \text{dbinom}(k, n, p)^j \cdot \text{pbinom}(k - 1, n, p)^{m-j} \]

\[\text{FEC}(n, m, p) := \sum_{i=1}^{n} i \cdot \text{pdfFEC}(n, m, p, i) \]
Example Message Reduction

Block=20 packets, Group Sizes of 20, 50, 100

Joe Macker March 1996